



S7-1200

Instrukcje skalowania i normalizacji, sterujące wykonywaniem programu, oraz służące do przesuwania i obrotu danych

W EP5/2010 przedstawiliśmy pierwszą część opisu instrukcji obsługiwanych przez sterowniki S7-1200, w EP6/2010 zajęliśmy się przybliżeniem instrukcji umożliwiających wykonywanie stało- i zmiennoprzecinkowych operacji arytmetycznych. W tym artykule skupiamy się na prezentacji i omówieniu instrukcji służących do skalowania, normalizacji, sterowania przebiegu programu, a także instrukcji służących do przesuwania i obrotu.

Instrukcje skalowania i normalizacji

SCALE_X skaluje znormalizowany parametr VALUE ($0.0 \leq \text{VALUE} \leq 1.0$) do typu danej i zakresu wartości wyspecyfikowanych przez parametry MIN i MAX

$$\text{OUT} = \text{VALUE} (\text{MAX} - \text{MIN}) + \text{MIN}$$

NORM_X normalizuje parametr VALUE wewnątrz zakresu wartości wyspecyfikowanych przez parametry MIN i MAX.

elektronikaB2B.pl

$OUT = (VALUE - MIN)/(MAX - MIN)$,
gdzie $(0.0 \leq OUT \leq 1.0)$

LAD		FBD	
SCALE_X Real to ??? EN ENO MIN OUT VALUE MAX		SCALE_X Real to ??? EN ENO MIN OUT VALUE MAX	
NORM_X ??? to Real EN ENO MIN OUT VALUE MAX		NORM_X ??? to Real EN MIN OUT VALUE MAX	

Parametr	Typ danych	Opis
MIN	SINT, INT, DINT, USINT, UINT, UDINT, REAL	Minimalna wartość wejściowa zakresu
VALUE	SCALE_X: REAL NORM_X: SINT, INT, DINT, USINT, UINT, UDINT, REAL	Wartość wejściowa do skalowania lub normalizacji
MAX	SINT, INT, DINT, USINT, UINT, UDINT, REAL	Maksymalna wartość wejściowa zakresu
OUT	SCALE_X: SINT, INT, DINT, USINT, UINT, UDINT, REAL NORM_X: REAL	Przeskalowana lub znormalizowana wartość wyjściowa

Parametr VALUE dla SCALE_X powinien być ograniczony do zakresu $(0.0 \leq VALUE \leq 1.0)$.

Jeżeli parametr VALUE jest mniejszy od 0.0 lub większy od 1.0, to:



- W wyniku liniowej operacji skalowania otrzymane wartości OUT mogą być mniejsze niż parametr MIN lub większe niż parametr MAX, ale zgodne z zakresem wartości danych typu ustalonego dla OUT. W takich przypadkach po wykonaniu operacji SCALE_X ustawiany jest ENO = TRUE.

- Możliwe jest, że przeskalowane liczby nie zawierają się w dopuszczalnym zakresie danych typu ustalonego dla OUT. W takich przypadkach wartość parametru OUT przyjmuje wartość pośrednią równą najmniej znaczącej części rzeczywistej liczby skalowanej przed wykonaniem końcowej konwersji na daną typu OUT. Wówczas po wykonaniu operacji SCALE_X ustawiany jest ENO = FALSE.

Parametr VALUE dla NORM_X powinien być ograniczone do zakresu $(MIN \leq VALUE \leq MAX)$.

Jeżeli parametr VALUE jest mniejszy niż MIN lub większy niż MAX, to operacja liniowego skalowania może w wyniku dać znormalizowaną wartość OUT, która jest mniejsza od 0.0 lub większa od 1.0. W takim przypadku po wykonaniu operacji SCALE_X ustawiany jest ENO = TRUE.

Kody warunkowe

STATUS ENO	Warunek	Wynik OUT
1	Brak błędu	Ważny wynik
0	Wynik wykracza poza ważny zakres dla danych typu OUT	Wynik pośredni: najmniej znacząca część rzeczywistej liczby skalowanej przed wykonaniem końcowej konwersji na daną typu OUT.
0	Parametry MAX \leq MIN	SCALE_X: najmniej znacząca część liczby rzeczywistej VALUE wypełnia przestrzeń OUT.
0	Parametr VALUE = \pm INF lub \pm NAN	VALUE jest wpisywany do OUT.

Instrukcje skoków i etykiety

Instrukcje sterujące wykonaniem programu są wykorzystywane do warunkowego wykonywania ciągu operacji:

JMP: Jeżeli do cewki JMP dochodzi zasilanie (LAD) lub na wejściu ramki jest wartość TRUE (FBD), to program jest kontynuowany od pierwszej instrukcji po wyspecyfikowanej etykiecie.

JMPN: Jeżeli do cewki JMP nie dochodzi zasilanie (LAD) lub na wejściu ramki jest wartość FALSE (FBD), to program jest kontynuowany od pierwszej instrukcji po wyspecyfikowanej etykiecie.

Label: Docelowa etykieta dla instrukcji skoku JMP i JMPN.

LAD	FBD
Label_name —(JMP)—	Label_name JMP
Label_name —(JMPN)—	Label_name JMP
Label_name	Label_name

Nazwy etykiet nadaje się poprzez bezpośrednie wpisywanie nazwy w instrukcji LABEL. Nazwę etykiety dla instrukcji JMP i JMPN wybiera się spośród dostępnych nazw używając ikony pomocy dla parametrów. Można również bezpośrednio wpisać nazwę etykiety do instrukcji JMP lub JMPN. Instrukcję RET stosuje się do zakończenia wykonywania bieżącego bloku.

LAD	FBD
Return_Value —(RET)—	Return_Value RET

Parametr	Typ danych	Opis
Return_Value	BOOL	Parametr „Return_value” instrukcji RET jest w bloku wywoływanym przypisywany parametrowi wyjściowemu ENO ramki wywołującej blok.

Opcjonalną instrukcją RET wykorzystuje się do zakończenia wykonywania bieżącego bloku. Zakończenie wykonywania bieżącego bloku nastąpi w tym miejscu i instrukcje znajdujące się za instrukcją RET nie będą wykonywane wtedy i tylko wtedy jeżeli na wejściu cewki RET znajduje się zasilanie (LAD) lub na wejściu ramki RET jest stan TRUE (FBD). Jeżeli bieżącym blokiem jest OB., to parametr „Return_Value” jest ignorowany. Jeżeli bieżącym blokiem jest FC lub FB, to wartość parametru „Return_Value” jest zwracana do procedury wywołującej jako wartość ENO ramki wywołującej.

Użytkownik nie ma obowiązku wpisywania instrukcji RET jako ostatniej instrukcji bloku; jest to wykonywane automatycznie. W pojedynczym bloku może występować wiele instrukcji RET.

Wejście EN ramki FC bloku kodu MAIN musi mieć wartość TRUE aby nastąpiło rozpoczęcie wykonywania FC.

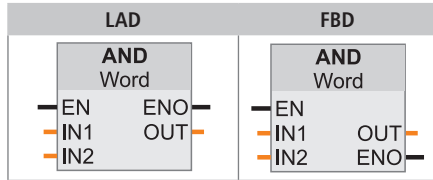
Po wykonaniu FC, w którym zasilanie na wejściu instrukcji RET ma wartość TRUE, na wyjściu ENO ramki FC kodu bloku MAIN pojawi się wartość wyspecyfikowana przez instrukcję RET w FC.

Instrukcje AND, OR i XOR

AND: logiczna operacja AND dla danych typu BYTE, WORD i DWORD

OR: logiczna operacja OR dla danych typu BYTE, WORD i DWORD

XOR: logiczna operacja XOR dla danych typu BYTE, WORD i DWORD



Parametr	Typ danych	Opis
IN1, IN2	BYTE, WORD, DWORD	Wejścia logiczne
OUT	BYTE, WORD, DWORD	Wyjście logiczne

Podczas wyboru typu danych jest ustalany taki sam typ danych parametrów IN1, IN2 i OUT. Odpowiadające sobie bity IN1 i IN2 są argumentami operacji logicznej, której wynik jest wpisywany do OUT. Po zakończeniu wykonywania powyższych instrukcji, ENO ma zawsze wartość TRUE.

Instrukcja inwersji

Instrukcja INV jest stosowana do wyznaczenia dwójkowego uzupełnienia do jedności parametru IN. Uzupełnienie do jedności jest wykonywane poprzez inwersję każdego bitu parametru IN (zamianę każdego 0 na 1 i 1 na 0). Po zakończeniu wykonywania instrukcji, ENO ma zawsze wartość TRUE.

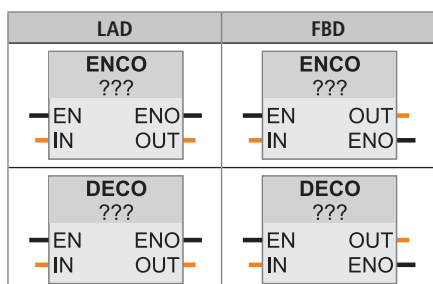


Parametr	Typ danych	Opis
IN	SINT, INT, DINT, USINT, UINT, UDINT, BYTE, WORD, DWORD	Element podlegający inwersji
OUT	SINT, INT, DINT, USINT, UINT, UDINT, BYTE, WORD, DWORD	Wyjście po inwersji

Instrukcje enkodowania i dekodowania

Polecenie ENCO koduje ciąg bitów na liczbę dwójkową.

Polecenie DECO dekoduje liczbę dwójkową na ciąg bitów.



Parametr	Typ danych	Opis
IN	ENCO: BYTE, WORD, DWORD DECO: UINT	ENCO: ciąg bitów do zakodowania DECO: wartość do dekodowania
OUT	ENCO: INT DECO: BYTE, WORD, DWORD	ENCO: wartość zakodowana DECO: ciąg bitów zdekodowany

Instrukcja ENCO dokonuje konwersji parametru IN na liczbę dwójkową odpowiadającą ciągowi bitów znajdujących się na najmniej znaczącej pozycji w IN i zapisuje wynik jako parametr OUT. Jeżeli parametr IN jest równy 0000 0001 albo 0000 0000, to do OUT jest wpisywane 0. Jeśli parametr IN ma wartość 0000 0000 to ENO przyjmuje wartość FALSE.

Instrukcja DECO dekoduje liczbę dwójkową z parametru IN poprzez ustawianie bitów znajdujących się na odpowiadających pozycjach OUT na 1 (wszystkie pozostałe bity są ustawione na 0). Po zakończeniu wykonywania instrukcji, ENO ma zawsze wartość TRUE.

Wybór typu danej parametru OUT instrukcji DECO spośród BYTE, WORD lub DWORD ogranicza użyteczny zakres parametru IN. Jeżeli wartość parametru IN przekracza użyteczny zakres, to wykonywana jest pokazana poniżej operacja modulo pozwalająca wydobyc najmniej znaczące bity.

Zakres parametru IN instrukcji DECO:

3 bity (wartość 0 – 7) IN są wykorzystywane do ustalenia 1 pozycji bitu w bajcie OUT.

4 bity (wartość 0 – 15) IN są wykorzystywane do ustalenia 1 pozycji bitu w słowie OUT.

5 bitów (wartość 0 – 31) IN jest wykorzystywanych do ustalenia 1 pozycji bitu w podwójnym słowie OUT.

Wartość IN instrukcji DECO		Wartość OUT instrukcji DECO (dekodowanie pojedynczej pozycji bitu)	
		Typ OUT: BYTE (8 bitów)	
Min. IN	0	00000001	
Maks. IN	7	10000000	
		Typ OUT: WORD (16 bitów)	
Min. IN	0	0000000000000001	
Maks. IN	15	1000000000000000	
		Typ OUT: DWORD (32 bity)	
Min. IN	0	000000000000000000000001	
Maks. IN	31	100000000000000000000000	

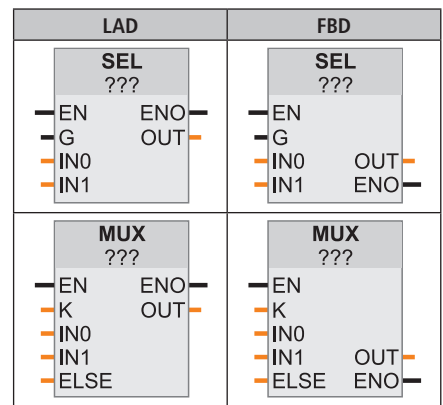
Kody warunkowe dla ENCO

STATUS ENO	Warunek	Wynik (OUT)
1	Brak błędu	Ważny wynik
0	IN jest zerem	OUT ustawiony na zero

Instrukcje wyboru (SEL) i multipleksowania (MUX)

SEL, w zależności od wartości parametru G, przypisuje jedną z dwóch wartości wejściowych parametrowi OUT.

MUX, w zależności od wartości parametru K, przypisuje jedną z wielu wartości wejściowych parametrowi OUT. Jeżeli parametr K wykracza poza dozwolony zakres, to parametrowi OUT jest przypisywana wartość parametru ELSE.



Parametry dla SEL

Parametr	Typ danych	Opis
G	BOOL	Przełącznik selektora: FALSE dla IN0 TRUE dla IN1
IN0, IN1	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wejścia
OUT	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wyjście

Parametry dla MUX

Parametr	Typ danych	Opis
K	UINT	Wartość selektora: 0 dla IN0 1 dla IN1 ...
IN0, IN1, ...	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wejścia

ELSE	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wartość wejściowa dla podstawienia (opcjonalnie)
OUT	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wyjście

Zmienne wejściowe i zmienna wyjściowa muszą być tego samego typu danych.

Instrukcja SEL zawsze wybiera pomiędzy dwoma wartościami wejściowymi.

Instrukcja MUX wybrana po raz pierwszy w edytorze programu ma dwa parametry wejściowe IN, ale można ją rozszerzyć dodając więcej parametrów IN.

Do dodawania i usuwania parametrów wejściowych instrukcji MUX służy następująca metoda:

W celu dodania wejścia należy kliknąć prawym klawiszem myszy na końcówkę wejściową jednego z istniejących parametrów IN i wybrać komendę „insert input”.

W celu usunięcia wejścia należy kliknąć prawym klawiszem myszy na końcówkę wejściową jednego z istniejących parametrów IN (jeżeli jest więcej wejść niż oryginalne dwa) i wybrać komendę „Delete”.

Kody warunkowe

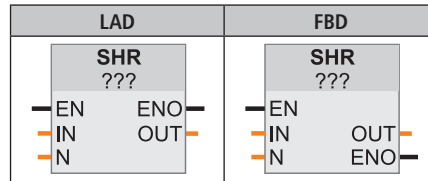
Po zakończeniu wykonywania instrukcji SEL, ENO ma zawsze wartość TRUE.

STATUS ENO (MUX)	Warunek	Wynik (OUT)
1	Brak błędu	Ważny wynik
0	K jest większe lub równe liczbie parametrów IN	Bez parametru ELSE: OUT nie ulega zmianie Z parametrem ELSE: OUT przyjmuje wartość ELSE

Instrukcje przesunięcia

Instrukcja przesunięcia (*shift*) jest stosowana do wzoru bitów parametru IN. Wynik jest przypisany parametrowi OUT. Parametr N określa o ile pozycji bitów ma nastąpić przesunięcie.

SHR: przesunięcie wzoru bitów w prawo.
SHL: przesunięcie wzoru bitów w lewo.



Parametr	Typ danych	Opis
IN	BYTE, WORD, DWORD	Wzór bitów do przesunięcia
N	UINT	Liczba pozycji bitów do przesunięcia
OUT	BYTE, WORD, DWORD	Wzór bitów po operacji przesunięcia

Dla N = 0 nie jest wykonywane przesunięcie i do OUT jest przypisywana wartość IN.

Na pozycje opróżnione podczas przesuwania są wpisywane zera.

Jeżeli liczba pozycji do przesunięcia (N) przekracza liczbę bitów wartości docelowej (8 dla BYTE, 16 dla WORD i 32 dla DWORD), to oryginalna wartość zniknie i zostanie zastąpiona przez zera (do OUT zostaną wpisane same zera).

Dla operacji przesuwania, ENO ma zawsze wartość TRUE.

Przykład operacji SHL dla danych typu WORD: podczas przesuwania z lewej strony są wpisywane zera			
IN	1110 0010 1010 1101	Wartość OUT przed pierwszym przesunięciem	1110 0010 1010 1101
		Po pierwszym przesunięciu w lewo	1100 0101 0101 1010
		Po drugim przesunięciu w lewo	1000 1010 1011 0100
		Po trzecim przesunięciu w lewo	0001 0101 0110 1000

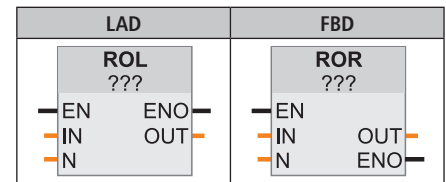
Instrukcje obrotu

Instrukcje obrotu są stosowane do cyklicznego przesuwania wzoru bitów para-

metru IN. Wynik jest przypisywany do parametru OUT. Parametr N określa o ile pozycji bitów ma nastąpić obrót.

Instrukcja ROR powoduje obrót wzoru bitów w prawo.

Instrukcja ROL powoduje obrót wzoru bitów w lewo.



Parametr	Typ danych	Opis
IN	BYTE, WORD, DWORD	Wzór bitów do obrotu
N	UINT	Liczba pozycji bitów do obrotu
OUT	BYTE, WORD, DWORD	Wzór bitów po operacji obrotu

Dla N = 0 nie jest wykonywany obrót i do OUT jest przypisywana wartość IN.

Podczas obrotu bity wysuwane z jednej strony trafiają na pozycje opróżniane z drugiej strony parametru docelowego; zatem żaden oryginalny bit nie jest tracony.

Jeżeli liczba pozycji do przesunięcia (N) przekracza liczbę bitów wartości docelowej (8 dla BYTE, 16 dla WORD i 32 dla DWORD), to obrót jest nadal wykonywany.

Dla operacji obrotu, ENO ma zawsze wartość TRUE.

Przykład operacji ROR dla danych typu WORD: podczas obrotu bity wysuwane z prawej strony trafiają na pozycje opróżniane z lewej strony			
IN	0100 0000 0000 0001	Wartość OUT przed pierwszym obrotem	0100 0000 0000 0001
		Po pierwszym obrocie w prawo	1010 0000 0000 0000
		Po drugim obrocie w prawo	0101 0000 0000 0000

Tomasz Starak

R E K L A M A

automatykaB2B.pl